1. **Describe Each of the following:**

   a. **Computer Graphics**

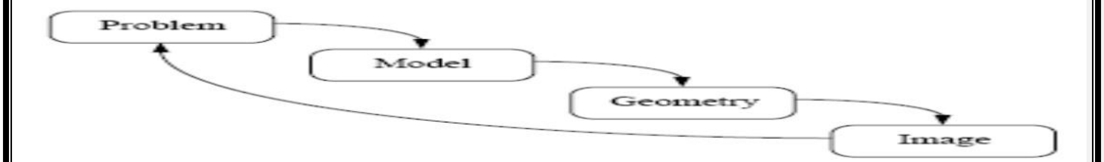
   An area of computer science and engineering that plays an important role in almost every application of computer software

   b. **Computer Graphics API**

   (APPLICATION PROGRAM INTERFACE)

   A set of tool that allow the programmer to design application using the concepts of the area

   c. **CG's can be used in solving Problems**



   d. **Graphics Pipeline**

   A special software/hardware subsystem that efficiently draws 3D primitives, and every desktop computer has a powerful **3D**

   e. **Video Memory**

   The portion of memory that is associated with the display

   f. **Memory Mapping**

   the status of each pixel on the screen was stored in a specific memory address

   g. **RGB Color Model**

   Each byte represents a number between (0) and (255) the degree of on primary color * Read – Green – Blue * (RGB – system)

h. **List of Common Computer Graphics Areas**

- Modeling
- Rendering
- Animation
- User interaction
- Virtual reality
- Image processing
- 3D scanning

i. **List of Common Computer Graphics Applications**

- Modeling
- Business
- Industry
- Government
- Computer ART
- Image processing
- CAD
- GUI

j. **Resolution**

Is the number of distinct pixels in each dimension that can be displayed without overlap on CRT

k. **Aspect Ratio**

Is the ratio of vertical points to horizontal points necessary to produce equal-length lines in both directions on the screen without light overlap

(ratio of horizontal points to vertical points)

l. **Persistence**

Is defined as the time it takes the emitted light from the screen to

Decay to 10% of its original intensity.

m. **Basic Components of computer graphics system**

A computer graphics system is a computer system; that have all the Components of a general-purpose computer system.

There are six major elements in our system :

1. Input devices
2. Central Processing Unit
3. Graphics Processing Unit
4. Memory
5. Frame buffer
6. Output devices

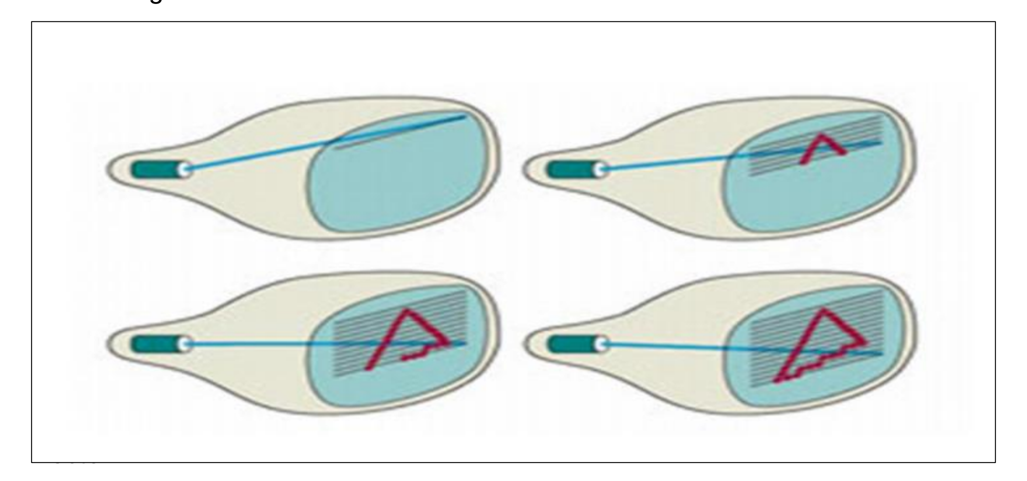
2. **Differentiate between raster scan and random scan system?**

Raster Scan Display Systems :

Line * columns

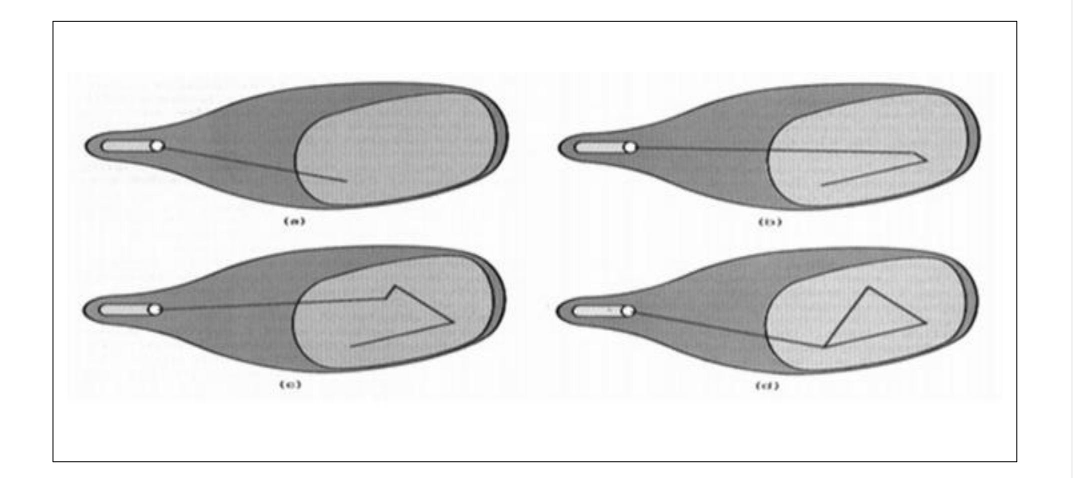
Example :

- A flat-panel computer display or TV
- Laser printers
- A digital camera



Example :

- A pen plotter



3. **Give an example of memory mapping for 5 X 5 resolution display device in case of monochrome system?**

Screen

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	`19	20
21	22	23	24	25

Memory

1 0ff	
2 off	
3 0ff	15 0ff
4 off	16 on
5 off	17 0ff
6 off	18 off
7 on	19 off
8 off	20 on
9 on	21 off
10 off	22 on
11 off	23 on
12 off	24 on
13 on	25 off
14 off	

4. **Give an example of memory mapping for 5 X 5 resolution display device in case of color system?**

Screen

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	`19	20
21	22	23	24	25

Memory

1 FF FF FF	
2 FF FF FF	
3 FF FF FF	15 FF FF FF
4 FF FF FF	16 00 FF 00
5 FF FF FF	17 FF FF FF
6 FF FF FF	18 FF FF FF
7 FF 00 00	19 FF FF FF
8 FF FF FF	20 00 FF 00
9 FF 00 00	21 FF FF FF
10 FF FF FF	22 00 FF 00
11 FF FF FF	23 00 FF 00
12 FF FF FF	24 00 FF 00
13 00 00 FF	25 FF FF FF
14 FF FF FF	

5. **What is the memory size required for storing image of height 7 and depth 10 in case of: (a) Monochrome display system (b) Color display system.**

6. **What are the common methods for storing and processing Images?**

Images file : A file that contains graphics data;

for example, a GIF or PNG file

Raster Image:

• The images is considered as rectangular arrays of pixels, each pixel have different colors stored as three numbers, for red, green, and blue.

• In a Monochrome system [black-and-white], each screen point is either on (a bit value of 1) or off (a bit value of 0), so only one bit per pixel is needed to store the intensity of screen positions.

• A system with 24 bits per pixel and a screen resolution of 1024 by 1024 requires 3 megabytes of storage for the frame buffer.

Vector Image:

• The image is stored as a set of instructions (set of line drawing instructions) for displaying the image (lines , shapes, areas) rather than the pixels needed to display it.

• Are often used for text, diagrams, mechanical drawings.

• Advantage:

• Resolution independent and can be displayed well on very high resolution devices.

• Disadvantage:

• They must be rasterized before they can be displayed.

7. **What are the popular image storage formats?**

   Popular image storage formats include:

   - Jpeg format
   - Tiff format
   - Ppm format
   - Png format

8. **What the difference between mathematical point and computer graphics point? And Mathematical line, computer graphics line?**

9. **What steps must a computer take to plot a point (22.25,10.4)?**

A computed line position of (**22.25,10.4**) [Mathematical Point], for example,

Would be converted to pixel position (**22, 10**). Thus rounding of coordinate values to integer’s causes lines to be displayed with a stair-step appearance, as represented

10. **List the operating characteristics for the video display systems based on the CRT technology.**

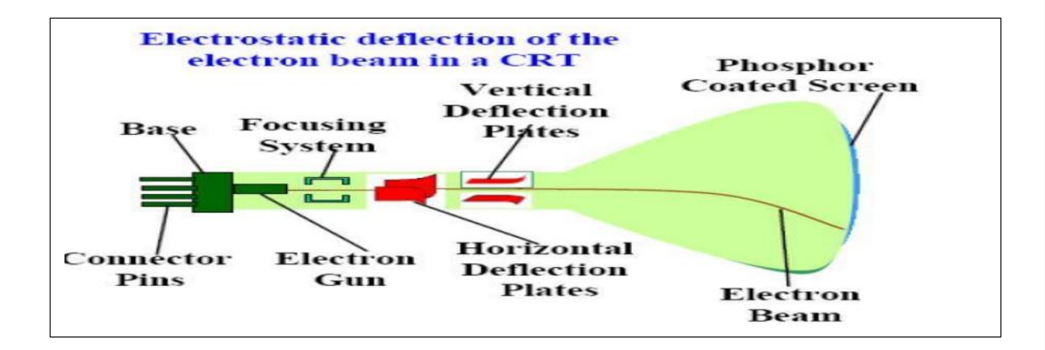
a) Connector pins
b) Focusing system
c) Defalcation plates
d) Phosphor coated surface in screen

Pixel value generate Binary to Analog,

وبعد كدا بيمعل force

وبيحدث اصتدام وينتج عنو اضائة ومع مرور الوقت بينتهي عشان كدا بنحتاج لعملية ال

Refers



11. **What are the different types of Flat-Screens, and what is the difference between them?**

جميعهم عباره عن (ARRY) بيولد مجال كهربائي

- LED panel: light-emitting diodes
- LCD display: polarization of the liquid crystals
- Plasma panel: energize gases

12. **Consider three different raster systems with resolutions of 640 by 480, 1280 by 1024, and 2560 by 2048. What size frame buffer (in bytes) is needed for each of these systems to store 12 bits per pixel? How much storage is required for each system if 24 bits per pixel are to be stored?**

Frame-buffer size for each of the systems is

• 640 × 480 × 12 bits ÷ 8 bits per byte = 450 KB

• 1280 × 1024 × 12 bits ÷ 8 bits per byte = 1920 KB

• 2560 × 2048 × 12 bits ÷ 8 bits per byte = 7680 KB

• For 24 bits of storage per pixel, each of the above values is doubled.

900 KB & 3840 KB & 15360 KB

13. **Suppose an RGB raster system is to be designed using an 8-inch by 10-inch screen with a resolution of 100 pixels per inch in each direction. If we want to store 6 bits per pixel in the frame buffer, how much storage (in bytes) do we need for the frame buffer?**

The size of frame buffer is (8 x 10 x 100 x 100 x 6)/ 8= 600000 bytes

14. **How long would it take to load a 640-by-480 frame buffer with 12 bits per pixel, if 105 bits can be transferred per second? How long would it take to load a 24-bit-per-pixel frame buffer with a resolution of 1280 by 1024 using this same transfer rate?**

Let X the time that will be taken to load a 640-by-480 frame buffer with 12 bits per
pixel.

Number of bits = 640 * 480 * 12 = 3686400 bits

1 sec   105 bits and X sec(s)   3686400 bits

Then X = 3686400/105 = 36.864 second

• Let X the time that will be taken to load a 1280 by 1024 frame buffer with 24 bits
per pixel.

Number of bits = 1280 * 1024 * 24= 31457280 bits

1 sec   105 bits and X sec(s)   31457280 bits

Then X = 31457280/ 105 = 314.5728 second

Describe how a point can be represented in display device?

- A random scan
- Black and white raster
- RGB raster system

1. **Define a graphical line and how it can be displayed on a specific display device?**
   - For the raster video display : the line color is loading into the frame buffer
   - After reading from the frame buffer the video controller then "plots" the screen pixels
   - Screen locations : **يتم الرجوع اليها مع القيم الصحيحة حتي المرسومة ويتم رسم خط تقريبي بين الاثنين المحديدين**
   - A computed line position : **(10 .48 , 20.15) يتم حساب موقف الخط مثلا**
   - **يتم تقريب الموضع الي البكسل (20 , 10) وهذا التقريب لتنسيق القيم الى الرقم الصحيح**
   - The characteristic stair-step : يتم تحسين دقة الخط
   - More effective techniques : **تقوم ع ظبط البكسل على طول مسارات الخط**

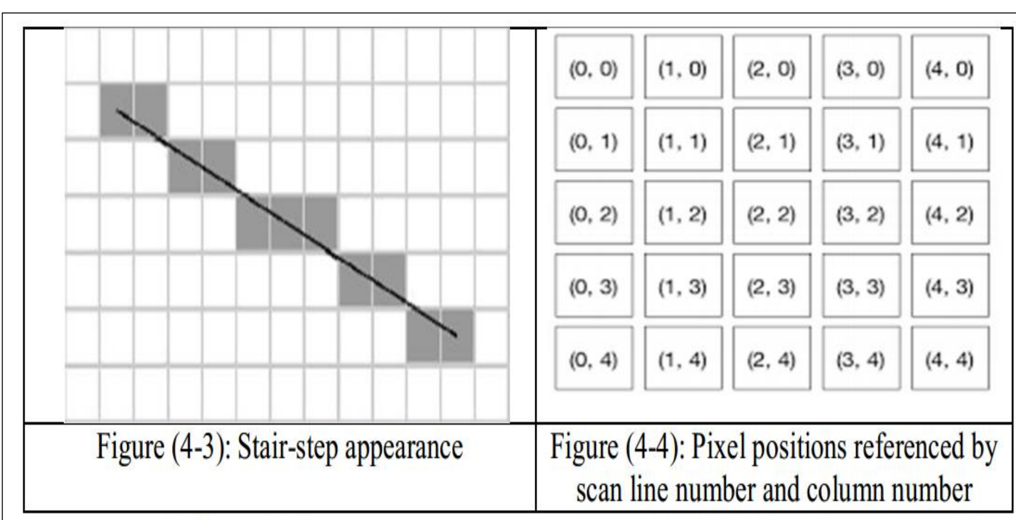


Figure (4-3): Stair-step appearance

Figure (4-4): Pixel positions referenced by scan line number and column number

- One of 2^N intensities or colors is associated with each pixel, where N is the number of bits per pixel.

A. Brute Force Algorithm

3. **Explain the Basic concept of drawing a line using the brute force algorithm?**

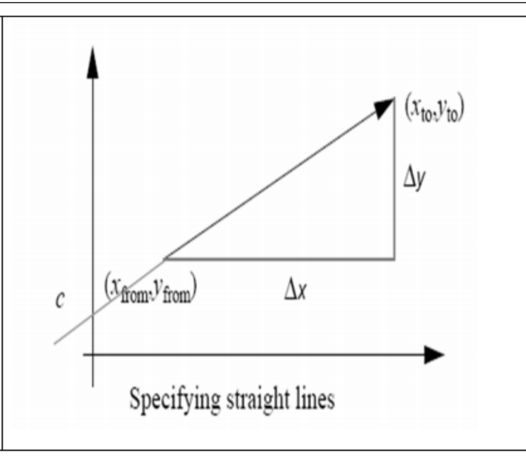
$$y = mx + c$$

where m is the gradient of the line:

$$m = \frac{\Delta y}{\Delta x} = \frac{y_{from} - y_{to}}{x_{from} - x_{to}}$$

and c is its intercept of the y-axis

$$c = y_{from} - m\, x_{from}$$



نفترض ان نهاية الخط معروفة للحصول على اي قيمة لل (   ) يمكننا حسابة عن طريق (   ) من خلال جمع قيمة ال   وال

4. **Write an algorithm for drawing line using brute force algorithm:**
   a. **Use X-axis as A counter.**
   b. **Use Y-Axis as a counter.**

```
Method lineBrute (xfrom, yfrom, xto, yto) {
    DeltaY = yto-yfrom;
    DeltaX = xto-xfrom;
    m = DeltaY/DeltaX;
    c=yfrom - (m*xfrom);
    for(int x= xfrom; x<xto; x++) {
        y= (m*x) + c;
        Plot(x, y)
} }//lineDraw
```

5. **What are the main disadvantages of the brute force algorithm and how can we solve it?**
    - Gaps started to be appeared
      الخط هيبقي مفرغ
    - It requires flatting point
      مش بيقبل الكسور
    - **حل الطريقة دي بأنو يفرض ان الميل اقل من ال (1)**
    - M **<** 1
    - **هيديني (   ) اما لو الميل اكبر من (1) هيديني (   )**

6. **Digitize a line from point (-1,-1) to point (4, 5) using brute force line generation algorithm?**

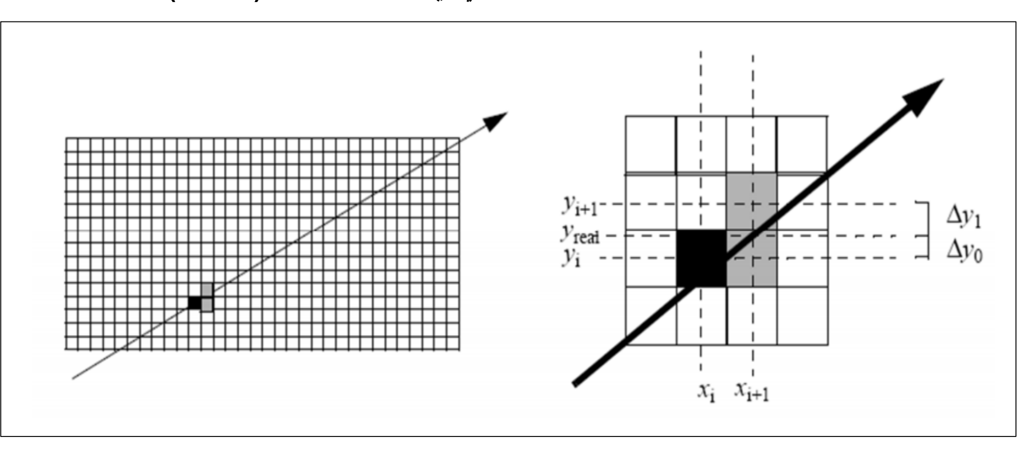
B. Bresenhams algorithm

7. **For the Bresenham's line drawing algorithm:**

   a. **Explain the basic concept of the Bresenham's algorithm?**

      Start by considering the simple case where 0 < m <1

      بيلغي اي حاجه فيها فلووت (الكسور)



   b. **Write the Algorithm**

Bresenham's Line-Drawing Algorithm for |m| < 1

1. Input the two line endpoints and store the left endpoint in (x_0, y_0)
2. Load (x_0, y_0) into the frame buffer; that is, plot the first point.
3. Calculate constants $\Delta x, \Delta y, 2\Delta y, and\ 2\Delta y - 2\Delta x$, and obtain the starting value for the decision parameter as $p_0 = 2\Delta y - \Delta x$
4. At each x_k along the line, starting at k = 0, perform the following test:

   If $p_k < 0$, the next point to plot is (x_{k+1}, y_k) and

$$p_{k+1} = p_k + 2\Delta y$$

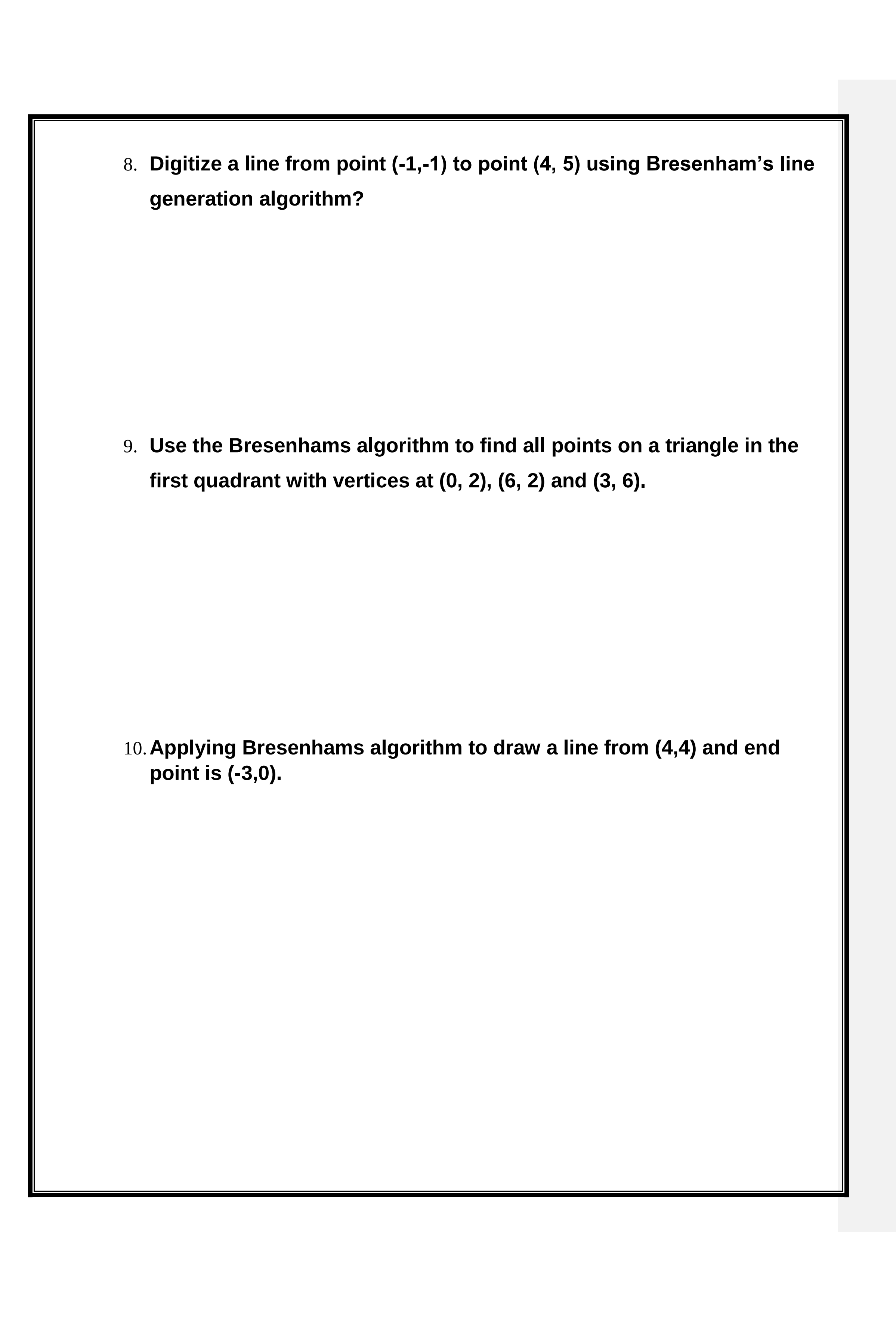
   Otherwise, the next point to plot is (x_{k+1}, y_{k+1}) and

$$p_{k+1} = p_k + 2\Delta y - 2\Delta x$$

5. Repeat step 4 Δx times.

c. **Write a c++ implementation for this algorithm.**

```
#Include <device.h>
void lineBresenham (int xa, int ya , int xb, int yb)
{
        int dx = abs ( xa - xb ) , dy = abs (ya - yb);
        int p = 2 * dy - d x ;
        int twoDy = 2 * dy, twoDyDx = 2 * (dy - Ax);
         int x , y, xEnd:
        /* Determine which point to use a s start, which as end * /
       if (xa > xb ) {
                    x = xb;                 Y = yb;               xEnd = xa;  }
       else{
                     x = xa;                Y = ya;               xEnd = xb;  }
                     setpixel (x, y);
                   while (x < xEnd) {
                         x++;
                           if (p < 0) { p += twoDy; }
                           else       { y++;       p+= twoDyDx; }
                            setpixel ( x , y);
              }
 }
```



8. **Digitize a line from point (-1,-1) to point (4, 5) using Bresenham's line generation algorithm?**

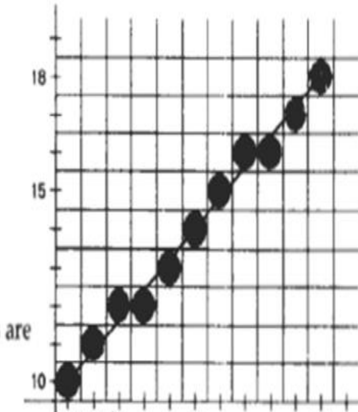
9. **Use the Bresenhams algorithm to find all points on a triangle in the first quadrant with vertices at (0, 2), (6, 2) and (3, 6).**

10. **Applying Bresenhams algorithm to draw a line from (4,4) and end point is (-3,0).**

11. **Write a Bresenham's line algorithm for line where |m| ≤ 1. Digitize a line with end points (20, 10) and (30, 18).**

To illustrate the algorithm, we digitize the line with endpoints (20, 10) and (30, 18). This line has a slope of 0.8, with

$$\Delta x = 10, \qquad \Delta y = 8$$



The initial decision parameter has the value

$$p_0 = 2\Delta y - \Delta x$$
$$= 6$$

and the increments for calculating successive decision parameters are

$$2\Delta y = 16, \qquad 2\Delta y - 2\Delta x = -4$$

We plot the initial point $(x_0, y_0) = (20, 10)$, and determine successive pixel positions along the line path from the decision parameter as

k	p_k	(x_{k+1}, y_{k+1})	k	p_k	(x_{k+1}, y_{k+1})
0	6	(21, 11)	5	6	(26, 15)
1	2	(22, 12)	6	2	(27, 16)
2	−2	(23, 12)	7	−2	(28, 16)
3	14	(24, 13)	8	14	(29, 17)
4	10	(25, 14)	9	10	(30, 18)

A plot of the pixels generated along this line path is shown in Fig.

Commented [M1]:

C. DDA Algorithm

12. **For the DDA line drawing algorithm:**

a. **Explain the basic concept of DDA?**

If m= 5 every change in (x) by 1

Result change in (y) by 5

b. **Write the Algorithm**

1. START
2. Get the values of the starting and ending co-ordinates i.e. , (x_1, y_1) and (x_2, y_2).
3. Find the value of slope m $m = \frac{\Delta y}{\Delta x} = \frac{y_2 - y_1}{x_2 - x_1}$
4. If $|m| \leq 1$ then $\Delta x = 1, \Delta y = m\Delta x$

   $x_k + 1 = x_k + 1, y_k + 1 = y_k + m$
5. If $|m| \geq 1$ then $\Delta y = 1, \Delta x = \Delta y / m$

   $x_k + 1 = x_k + 1/m, y_k + 1 = y_k + 1$
6. STOP

c. **Write a c++ implementation for this algorithm.**

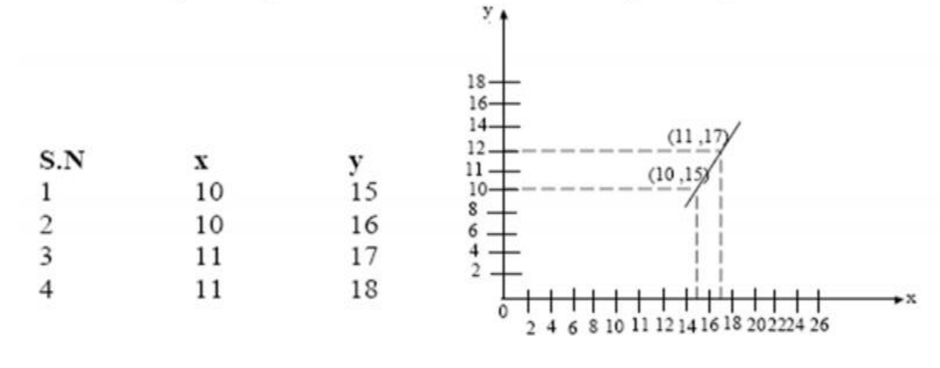
Eg. ***Example:*** *Digitize a line with end points (10,15) and (15,30).*

Solution:

The slope of line is $m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{30-15}{15-10} = \frac{15}{3} = 3$

$|m| > 1$

So we sample at y interval. The formula is given by $x_{k+1} = x_k + 1/m$.



S.N	x	y
1	10	15
2	10	16
3	11	17
4	11	18

13. **Using the DDA line drawing algorithm, find out the successive points that will be plotted by drawing a line (7,5) to (13,9)?**

14. **Consider a line from (0,0) to (6,7) Using simple DDA algorithm, digitize this line.**

15. **Use the DDA algorithm to find all points on a triangle in the first quadrant with vertices at (0, 2), (6, 2) and (3, 6).**

جزء ال دايرة لسه مش كامل وفيه حجات لسه ناقصه
ولو فيه حد عندو اي تعدلات ممكن يعرفني ع الميل بتاعي ع الفيس

M1.G@HOTMAIL.COM

يارب يكون الحاجه دي مفيدة للناس كلها ومش عايز منكو غير دعوه حلوة ليا ولكل الي بحبهم
وان شاء الله اي اضافات او اي تعديلات هعرف الكل

C. Circle Algorithm

1. **Write an algorithm for drawing the circle in each of following cases:**
   a. **Using the Cartesian Model $X^2+Y^2=R^2$ and use X-axis as a counter, and Center is (0,0).**

   b. **Using the Cartesian Model $X^2+Y^2=R^2$ and use Y-axis as a counter, and Center is (0,0).**

c. **Modify the point (a) so that the center is at the point (a,b).**

d. **Modify the point (b) so that the center is at the point (a,b).**

e. **Using the polar for representation of the circle (X= R cos □ , Y= R sin □) .**

f. **Modify the algorithm in point (e) and use the symmetry of the circle**

2. **Plot a circle at origin having center as (0,0) and radius=4 using the Cartesian Model $X^2+Y^2=R^2$. a. Use X axis as a counter.**

   b. Use Y Access as a counter.

3. **Plot a circle at origin having center as (5,5) and radius=3 using the Cartesian Model $(X-a)^2 + (Yb)^2 = R^2$.**

   a. **Use X-axis as a counter.**
   b. **Use Y-axis as a counter.**

4. **Plot a circle at origin having center as (0,0) and radius=3 using the Polar Form Model (X= R cos □**

   , Y= R sin □) .

   a. **Use X-axis as a counter.**
   b. **Use Y-axis as a counter.**

Problem 1

The DDA is an algorithm derived from the slope-intercept form of a line. From the DDA algorithm, which is first adapted for faster graphics, the Bresenham's algorithm is derived.

(a) Bresenham's has more characteristics than the DDA. List them.

Answer:

Some properties of Bresenham's algorithm from the textbook are:

1. **No rounding function**
2. **Only integer arithmetic**
3. **Calculation for the point (x_{i+1}, y_{i+1}) based on the point (x_i, y_i) only.**
4. **Applicable to the integer computation of circles**
5. **Line and integer circle algorithms provide the best-fit approximation**

Points 1 – 3 imply that the Bresenham's algorithm is faster than the DDA since rounding, floating arithmetic and non-incremental technique take more computing time.

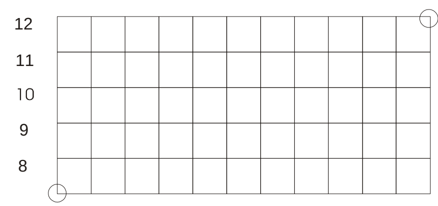
From the assignments of students, some points NOT the advantages of

Bresenham's algorithm over the DDA are: 1. endpoints are (x_0, y_0) and (x_1, y_1)

2. **slope is between 0 and 1.**
3. **description of the Bresenham's algorithm**
4. **no multiplication or division. (Both DDA and Bresenham's algorithm do not have these in the main loop.) 5. list of the variables used in the algorithms**

(b) Consider the Figure below where a line is to be placed on the grid from the circle in the lower left-hand corner to the circle in the upper right-hand corner by Bresenham's algorithm. Graphically show how Bresenham's algorithm will generate the line by making appropriate <u>grid-points</u>. For the

critical points, carry out simple calculations for the decision what needs to be done. Explain how you arrived at your answer.

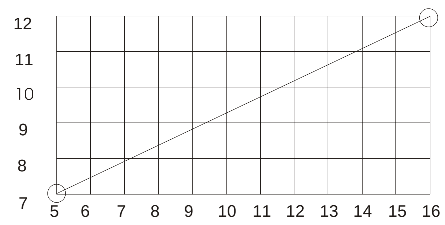


5 6 7 8 9 10 11 12 13 14 15 16

Answer:

The basic idea of the Bresenham's algorithm is that for a point P at (x_p, y_p) for a line of slope between 0 and 1, the point at x = x_p + 1 can be either E (y = y_p) or NE (y = y_{p+1}) depending on which one the line is close to.

To determine which points to turn on, a line is first drawn between the two end-points.



The line has slope $\frac{5}{11}$ and has equation $y = \frac{5}{11}x + b$ where b = $7 - \frac{25}{11} = \frac{52}{11}$.

Then, scan from left to right along the x-axis to decide which points to turn on based on the idea shown above.

For point at $x = 6$, y =7.45. Point (6,7) will be turned on. The points at x = 7 to x = 14 are very obvious. For point at x = 15, y = 11.55. Point (15,12) will be turned on.

Therefore, the result is:

